

## **ORGANIZING GRAPHICAL USER INTERFACES TO REVEAL HIDDEN AREAS**

### **BACKGROUND OF THE INVENTION**

#### **Field of the Invention**

[0001] This invention relates generally to the use and production of graphical user interfaces. More specifically, the invention relates to methods for arranging graphical user interfaces to reveal menus and other hidden areas which are not readily apparent to a user.

#### **Description of the Related Art**

[0002] As computers have become more sophisticated over the years, user interfaces to the computers and software that is run on the computers have also become more advanced. Graphical user interfaces (GUI) in particular have become very popular in providing users of computers, and particularly mouse input devices, with access to software, including menus for software options. GUIs are often provided in color, and color indications have long been used to indicate to users of GUIs what the particular state of a software function may be, and how to change these states.

[0003] In GUI systems, objects often have different methods that can be invoked using a "pull-down" menu. While the use of pull-down menus have been considered by those skilled in the art as logical and clear methods for displaying software menus, there are often hidden areas in the screen or choices that the user simply doesn't stumble across when attempting to manipulate the GUI to invoke the full scope of the software's functionality. Therefore, GUIs with hidden or masked option menus simply do not provide users with enough information to allow all of the versatility of a software program to be readily apparent and so oftentimes the software is not used to its fullest potential.

[0004] Figure 1 illustrates a prior GUI 10 which exhibits some of the aforementioned problems in the art. Besides a menu bar 20 on GUI 10 on which the user

might look for the different drop down menus that allow options for the software to be invoked, this GUI comprises three "option areas" 30, 40, 50 wherein a user of GUI 10 might click to access options. For example, option area 30 might comprise options for printing, saving, querying, or updating. Option area 40 might comprise options for printing, updating or resyncing with a host. Option area 50 might comprise options for printing, querying, or sending a document or data to a remote user. It will be appreciated by those with skill in the art that many other options for manipulating data and software may appear in a GUI such as that shown at 10 in Figure 1, and many types of software programs and/or regimes may employ such GUIs.

[0005] It will be further appreciated that a GUI 10 of the type shown in Figure 1 may be very poorly designed, especially with respect to option area 50 which is just a small sliver of the entire GUI window, or the display may be a size in pixels such that option area 50 is barely visible. Suppose that it is desired to invoke the option "send to a remote user" which is available in the option area 50. Since option area 50 is almost completely obscured in the current GUI 10 configuration, indeed the entire option area 50 could be scrolled entirely off of the window, the user may not be able to visually find this function. This results in the user spending a large amount of wasted time trying to find the option. This might include reading manuals, which are often cumbersome and inconclusive, or calling help desks to try and figure out where the option is or what might be the problem with the GUI 10.

[0006] Referring now to Figure 2, when the user attempts to resize the window in its hunt for the correct option, option area 30 has been greatly reduced in size, option area 40 has been reduced to nothing more than a sliver of the window, and option area 50 has totally disappeared. Therefore, in attempting to find the desired option by simply resizing the window, the user has completely lost the ability to find the option, and the GUI 10 is of no use in providing functional options to the user.

[0007] There is therefore a long-felt need in the art for methods of organizing GUIs so that software options and functions are readily apparent and easily findable. The

methods of the present invention should be readily adaptable to all types of software which utilize drop-down menus and other graphical displays for option picking. Moreover, the GUIs produced by the present invention should be easy to manipulate and operable for use with standard input devices such as a mouse or touch-sensitive capacitive pad. Such needs have not heretofore been fulfilled in the art.

### **SUMMARY OF THE INVENTION**

[0008] The aforementioned long-felt needs are met, and problems solved, by GUIs of the present invention. Preferably, the GUIs comprise at least one option area for displaying to a user functions which may be invoked by the user by operating an input device to the GUI. Even more preferably, an instrumentality is operable to be activated by the user when the user manipulates the input device for displaying at the user's command in a separate menu the functions that are displayable to the user in the at least one option area.

[0009] It will be appreciated that the instrumentality may comprise, for example, an icon that is visually displayed on the GUI or is otherwise conveniently viewed by the user when the user wishes to display the options in the separate menu. Alternatively, a series or combination of keystrokes that are enterable from a computer keyboard may be made operable to enable display of the separate menu with the options at the request of the user.

[0010] Additionally, the input device may be a mouse or other tactile device such as a capacitive touch pad. However, it will be appreciated that other input devices, for example, voice-activated input devices may also be used in conjunction with the present invention to accept input commands from the user. All such embodiments and equivalents thereof are intended to be within the scope of the present invention.

[0011] The present invention allows GUIs to function efficiently and expeditiously for users that are in need of invoking any software function that is offered by the software package. The inventive GUIs eliminate the need for users to